Optimizing Robotic Team Performance with Probabilistic Model Checking

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Sagar Chaki, Joseph Giampapa,
David Kyle (presenting), John Lehoczky

October 22nd, 2014

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
22 OCT 2014	**N/A**	

4. TITLE AND SUBTITLE
Optimizing Robotic Team Performance with Probabilistic Model Checking

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Kyle /David S.

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

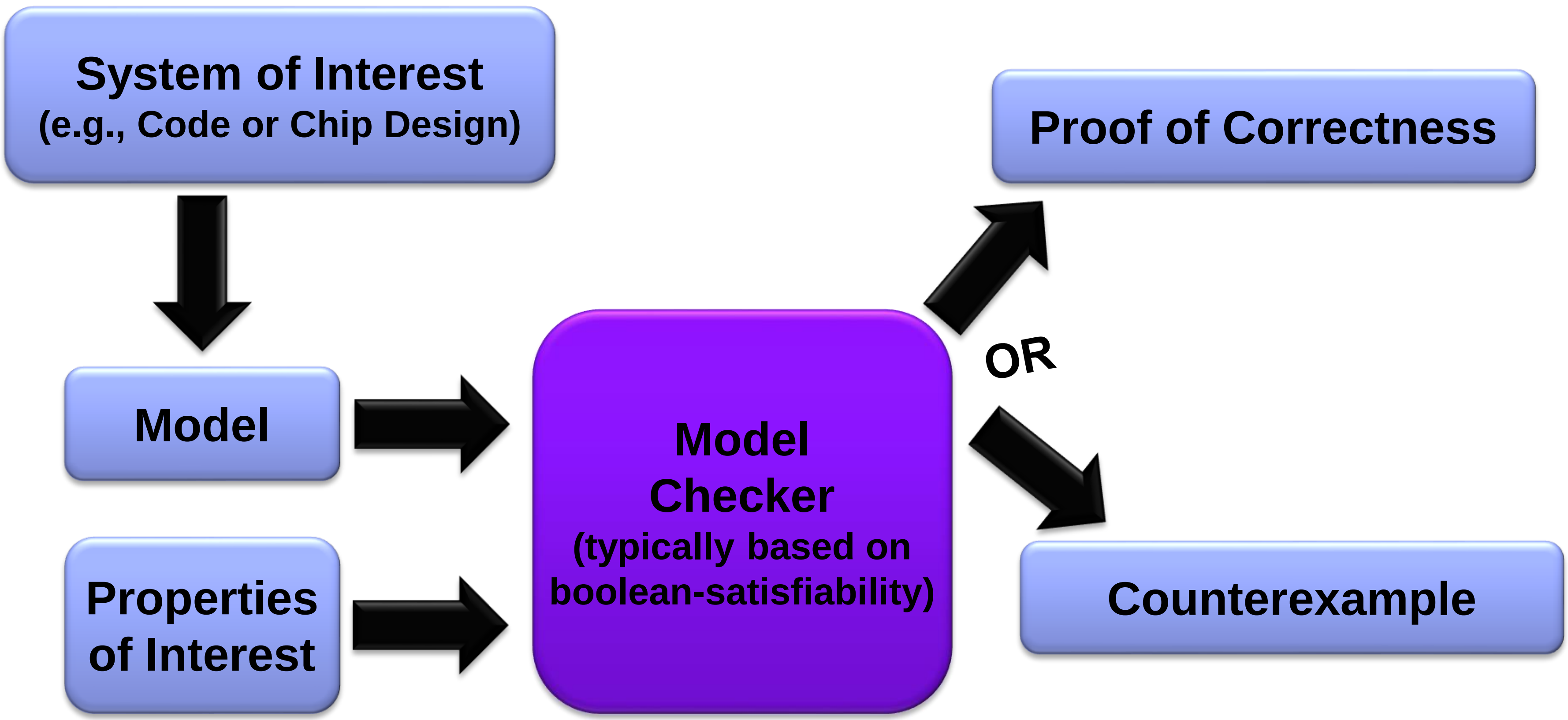
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**19**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Model Checking

Pentium floating point bug (1995): inspired Intel to model check chips

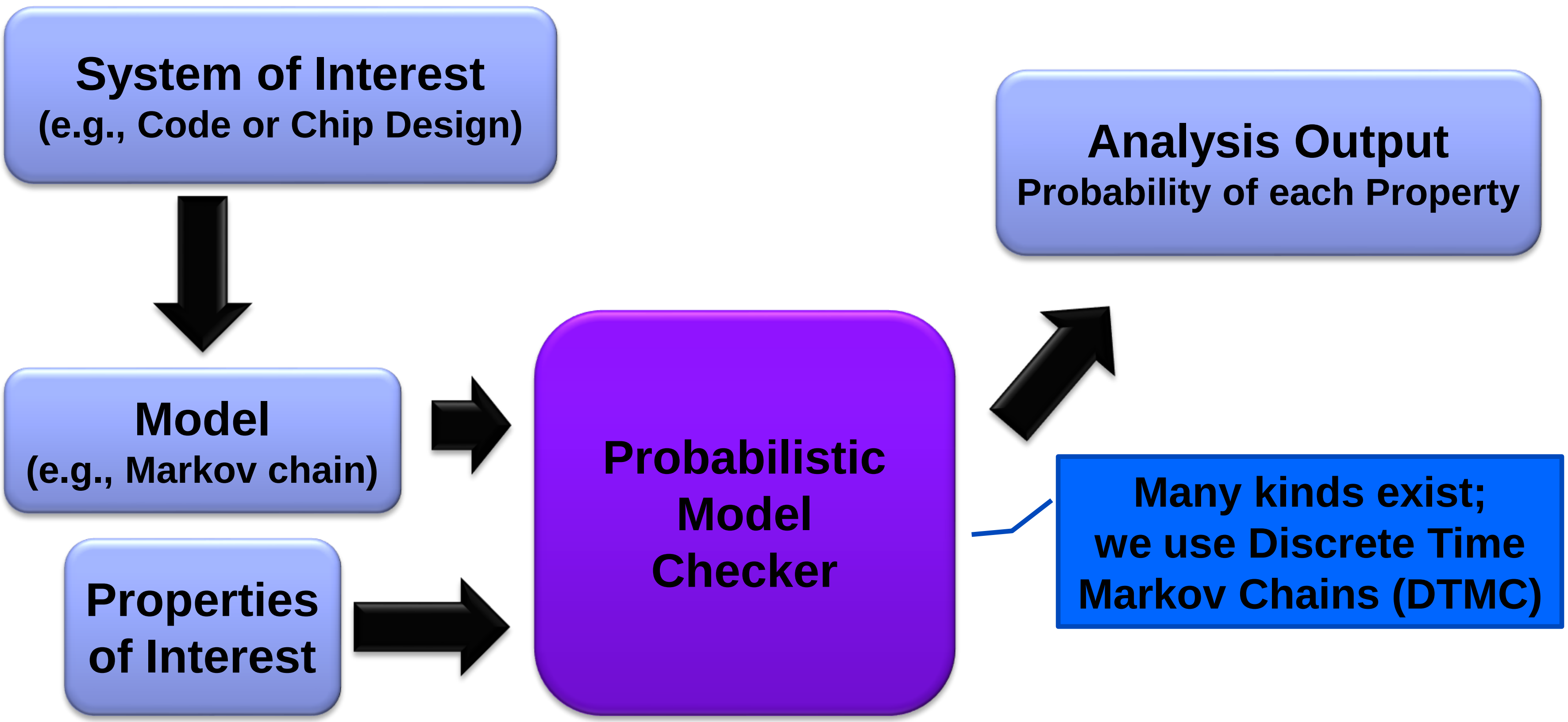
Now being applied to software, as well



Probabilistic Model Checking

Model Checking is purely boolean; a property is true or false.

For some systems, we want probabilities



DTMCs and Multi-Agent Robotic Systems

- Benefits:
  1. Performance vs physics-based simulation
  2. Exact results. Given a model, probabilities are calculated exactly

- Essential problems:
  1. Modelling physical systems is difficult
     – Can't just extract from a design or program code; must observe system to model it
     – Physical systems are continuous. Probabilistic Model Checking relies on discrete states
     – Given an imperfect model based on finite observations, how does that impact predictions?
  2. Robots interact. Modelling an entire system of multiple robots is hard.

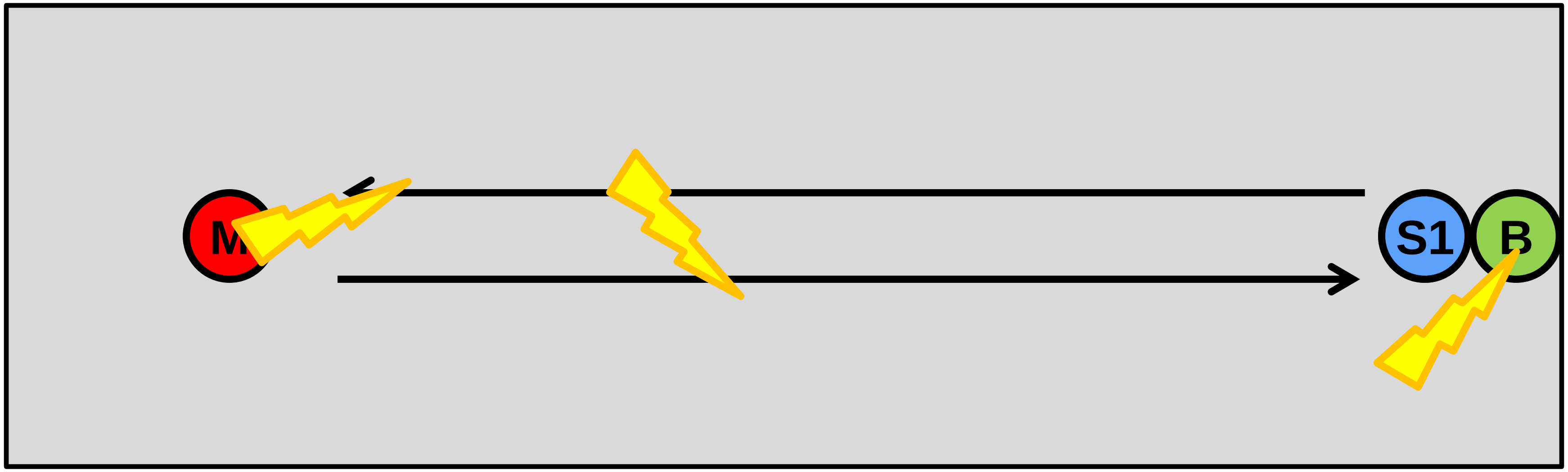
Our Contributions

1. Model robots individually:
   1. observe and measure individual behavior
   2. discretize observations in time and space, create Markov models
   3. compose these models into a Markov model of the whole system

2. Use known statistical error on the measurements made of the individual robots to produce estimates of error of the outputs of model checking the whole system.

Scenario

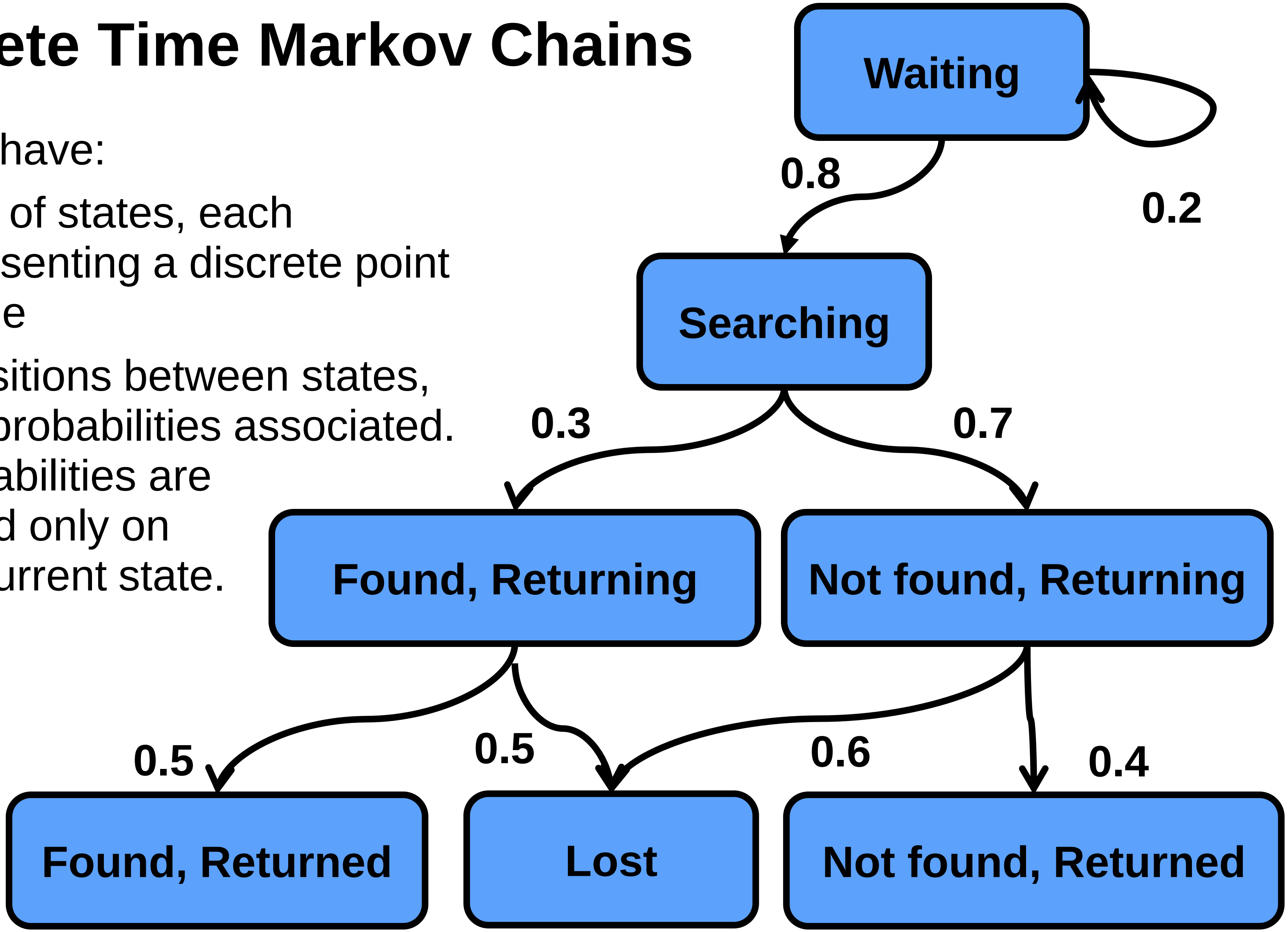
For our experiments, and as an illustration, we imagine a mine sweeping scenario. Objective: find a mine/IED in a constrained space (i.e., a drainage culvert under a road).



Discrete Time Markov Chains

DTMCs have:

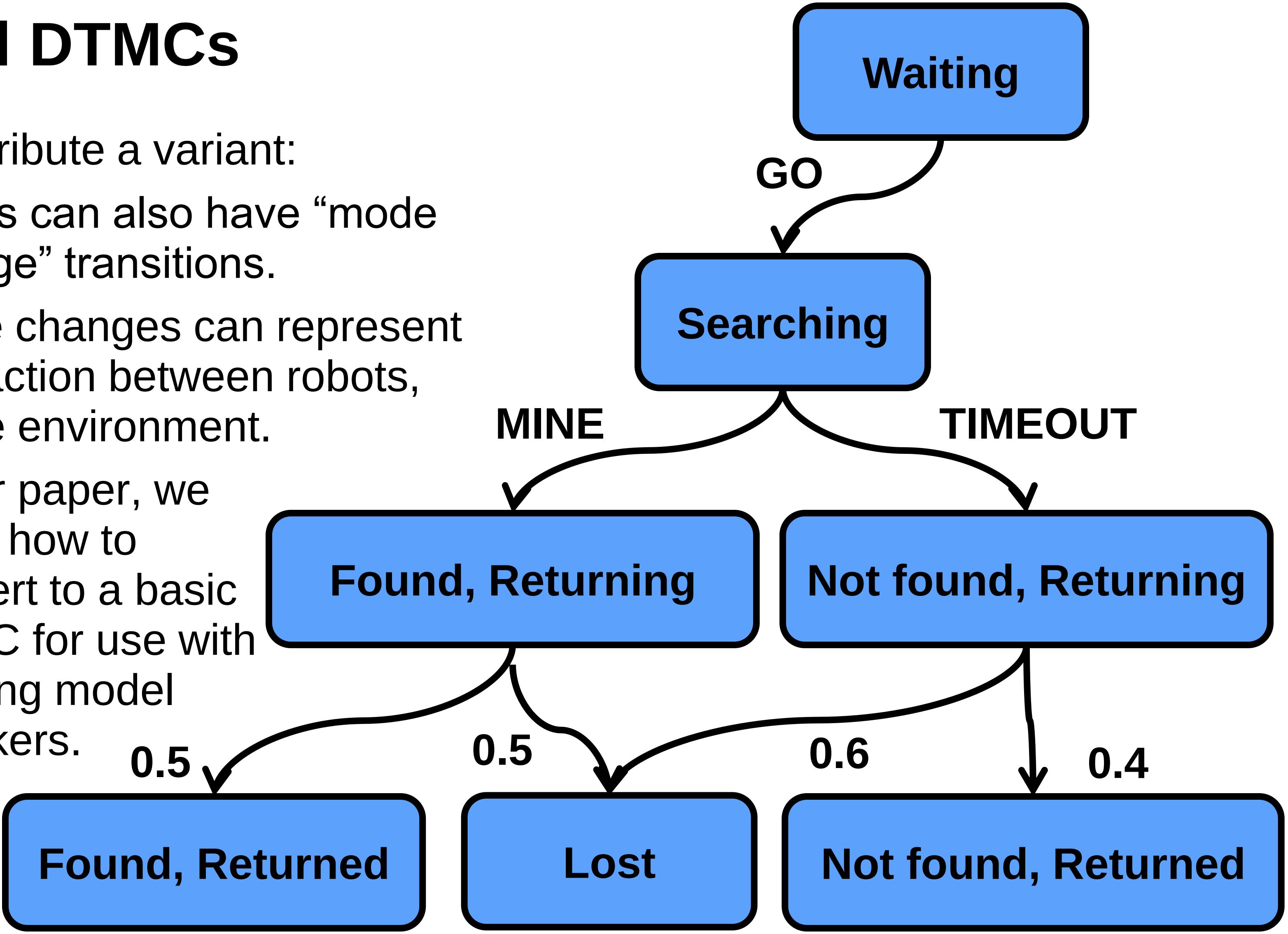
- A set of states, each representing a discrete point in time
- Transitions between states, with probabilities associated. Probabilities are based only on the current state.



Modal DTMCs

We contribute a variant:

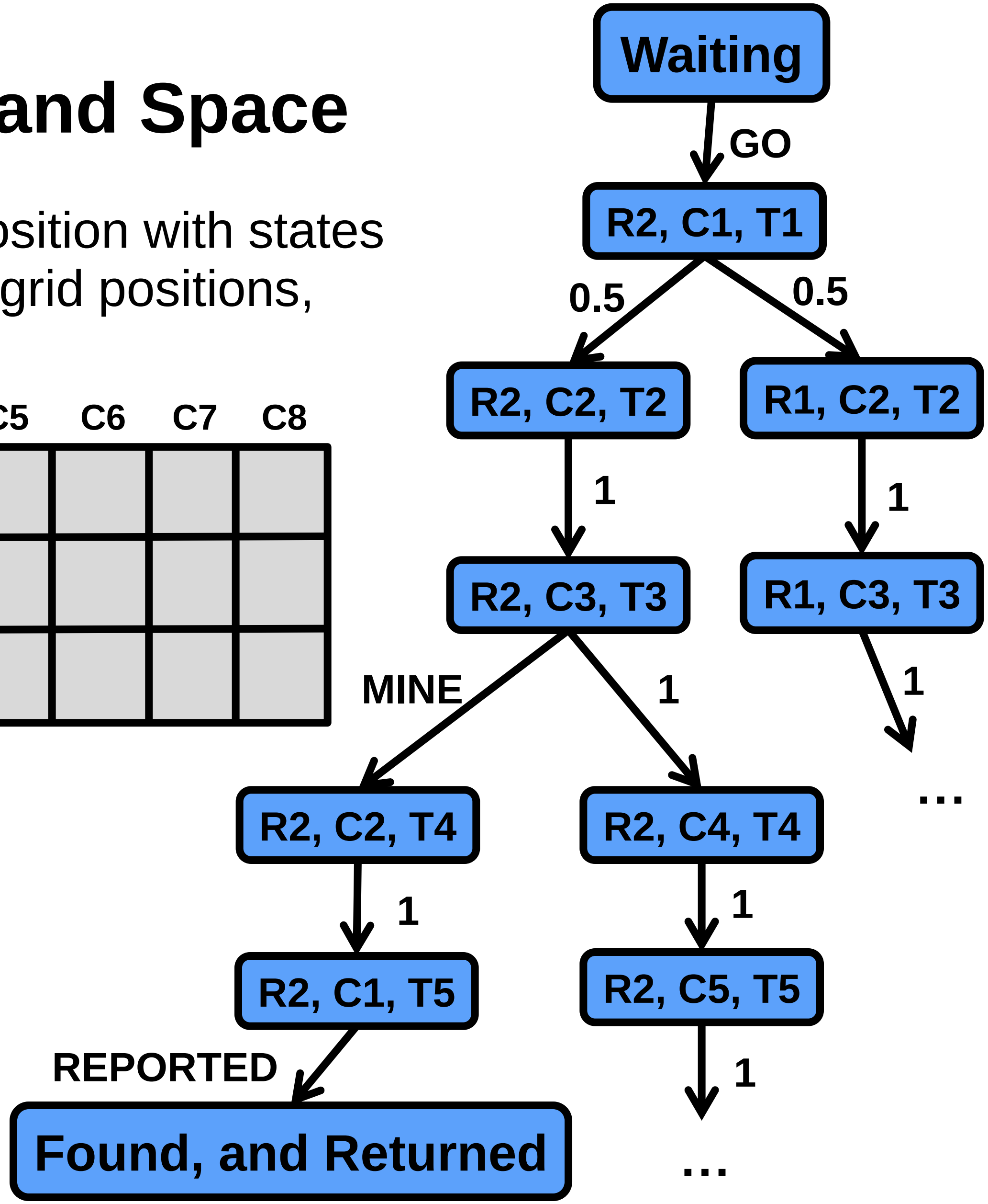
- States can also have “mode change” transitions.
- Mode changes can represent interaction between robots, or the environment.
- In our paper, we show how to convert to a basic DTMC for use with existing model checkers.



Discrete Time and Space

We represent robot position with states representing different grid positions, and different times.

	C1	C2	C3	C4	C5	C6	C7	C8
R1								
R2								
R3								



Composing Modal DTMCs

- Modal DTMCs allow us to model individual robots, then easily compose them together.

- To create an individual model:
  1. Run the robots individually, with pre-planned mode changes
  2. Observe the robot's behavior
  3. Create a Modal DTMC with transition probabilities based on observation, and mode changes as pre-planned

- Then, collect the individual modal DTMCs into a whole-system modal DTMC, and convert it to a non-modal DTMC

- Details of this construction, and correctness proof, are in the paper.

Error Estimation

- Probabilistic Model Checking itself has no error; given a model, it finds exact probabilities.

- However, modeling a robotic system will certainly not be perfect.

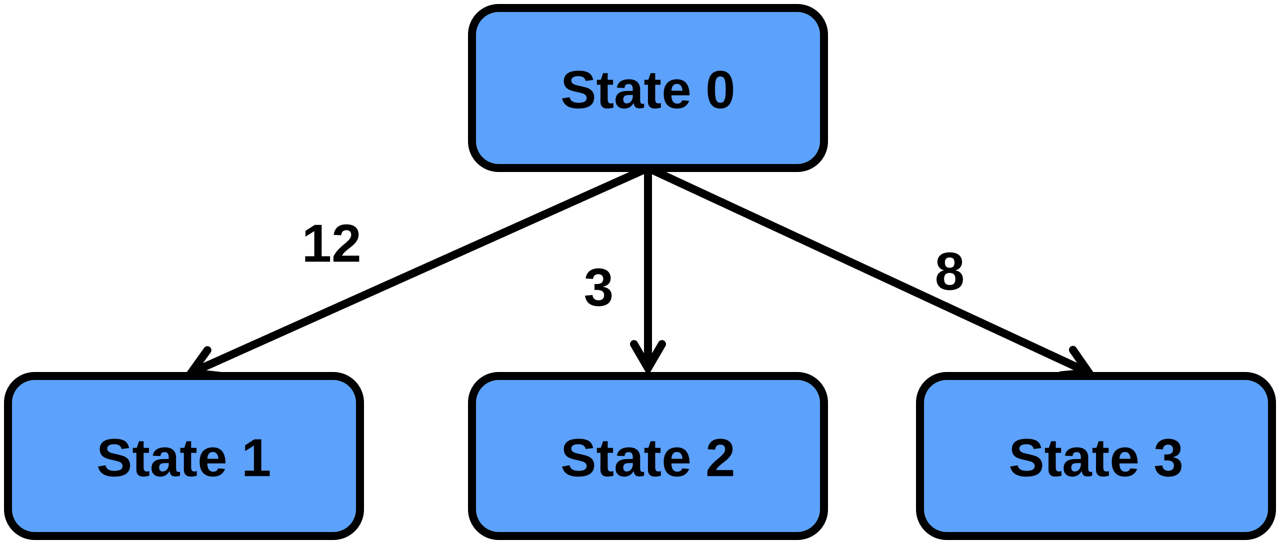
- Many kinds of error might appear causing a model to not reflect reality. We looked at handling one: the statistical errors due to observing the individual robots only a finite number of times.

- To examine this specific kind of error, we assume:
  - That the system can be fully described by a DTMC
  - That we have figured out the states of that DTMC
  - But the transition probabilities are observed over the course of finite trials

Dirichlet-based Distribution of DTMCs

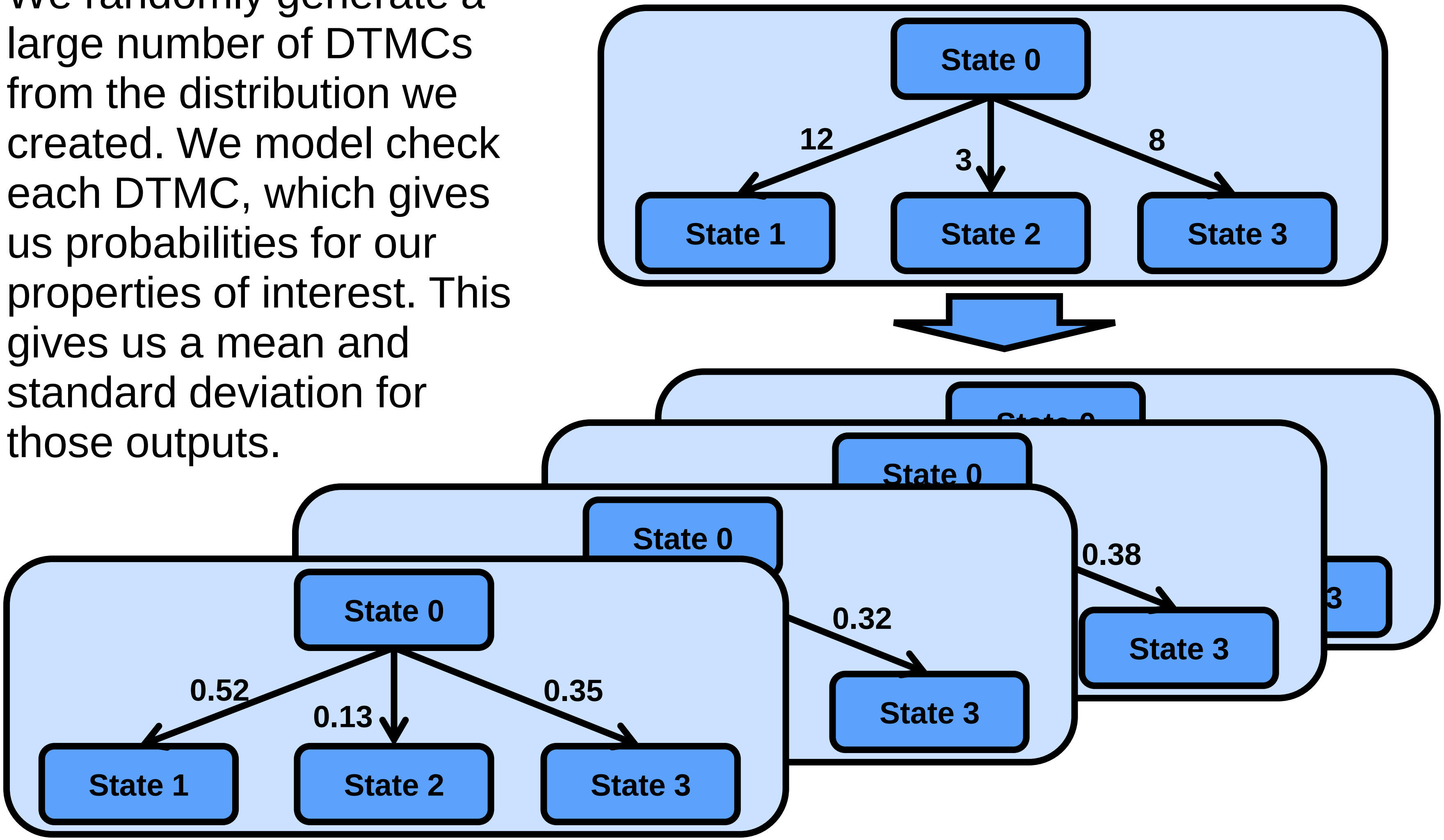
To analyze error, we create a random distribution of DTMCs.

For each transition, we use the counts of the times that transition was observed to describe the Dirichlet distribution of transition probabilities for that state, which includes a variance which shrinks with more observations.



Model Checking a distribution of DTMCs

We randomly generate a large number of DTMCs from the distribution we created. We model check each DTMC, which gives us probabilities for our properties of interest. This gives us a mean and standard deviation for those outputs.



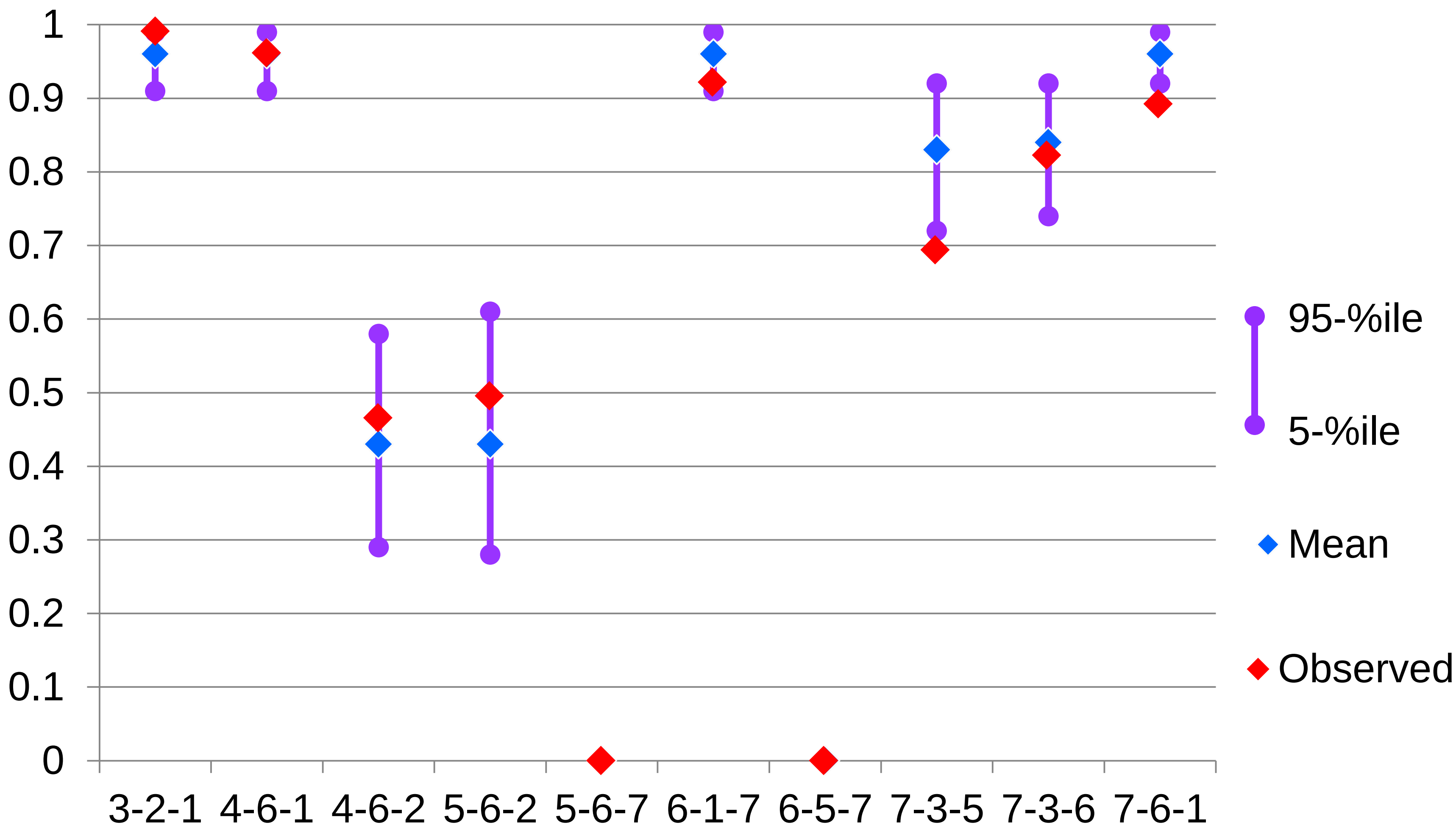
Experiment

- Used the simulator V-REP, with Kilobot models based on observations of real Kilobots

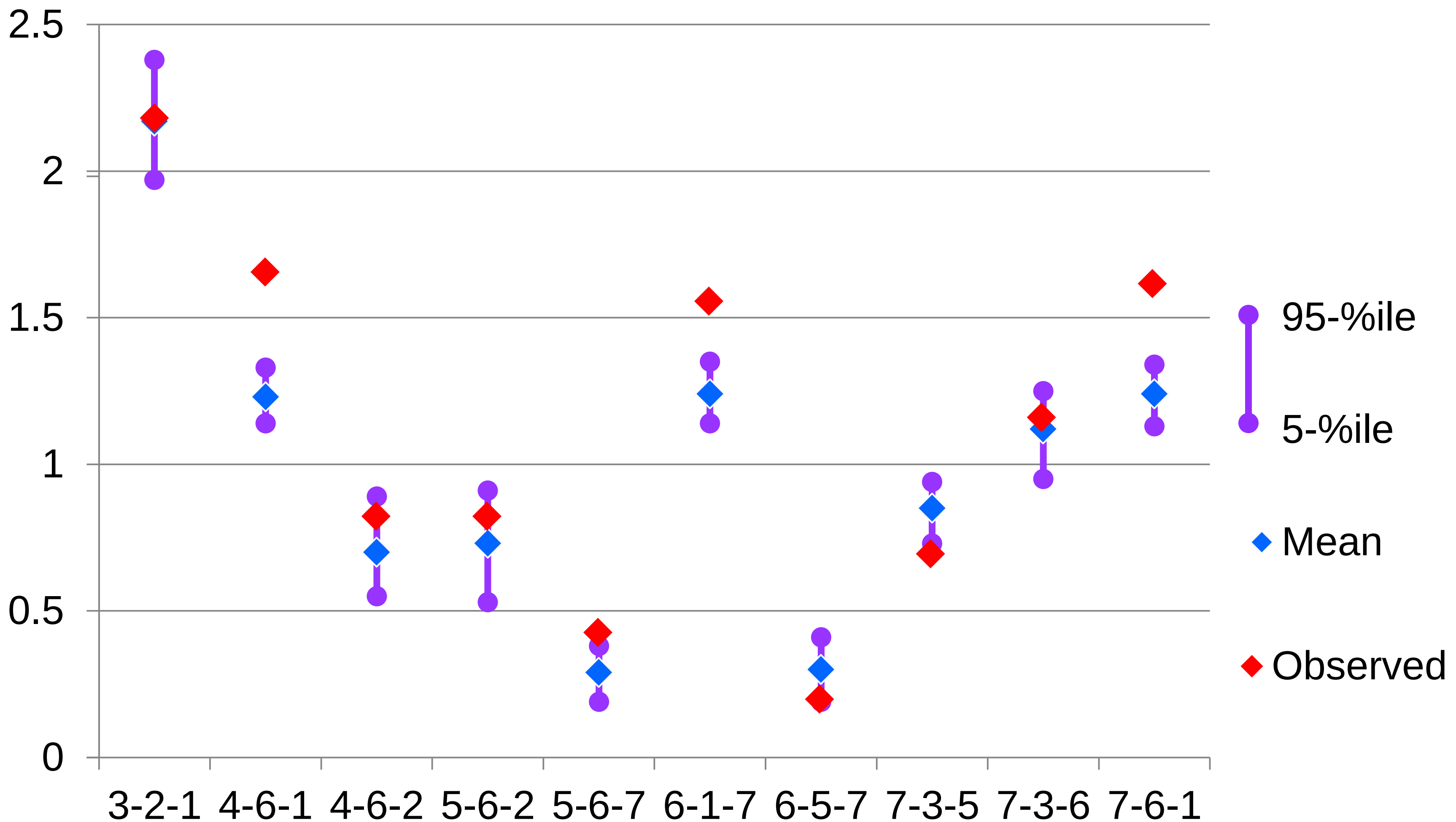
- Simulated Kilobots individually, used observations to create models for various team configurations (using Modal DTMCs), and predicted outcomes, using our Dirichlet sampling technique. Our metrics:
  - Probability base learns of mine (SUCCESS)
  - Expected number of bots that return to base (RETURNED)

- Simulated those teams in V-REP, and compared those outcomes to predicted outcomes.

Experiment Results – SUCCESS metric



Experiment Results – RETURNED metric



Questions?

Contact Information

Presenter
David Kyle
CPS/ULS initiative, DART project
Telephone: +1 412-268-9636
Email: dskyle@sei.cmu.edu

Software Engineering Institute
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web
www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

SEI Email: info@sei.cmu.edu
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257